REMARKS

Reconsideration is respectfully requested. Claims 1-10 and 12-15 are present in the application. Claims 12-15 are renumbered as claims 11-14. Claims 1-14 are amended herein.

Claims 1-10 and 12-15 (renumbered 1-14) are rejected under 35 U.S.C. §102(e) as allegedly being anticipated by Koselj et al (U.S. patent publication 2003/0214506). Applicants respectfully traverse.

Regarding claim 1, Koselj says that the invention therein relates to a generic display driver circuit, including a generic hardware graphics engine. Applicants' claims refer only to a specific hardware graphics engine circuit. For example:

a. Koselj Fig. 1, Fig. 2 and Fig 21 show only the text "Receive drawing commands (move, line, curve, fill)". Applicants' claims specify that the input data stream is an object data, which is different than the "commands" of Koselj. Object data as in applicants' claims, does not include commands (such as move, line, curve, fill), but only a list of object data (Bézier data list, color data list, color ramp, pattern or bitmap).

Applicants' claimed device, which can be implemented as a hardware circuit, does not process a command such as Curve, Move, Line or Fill because applicants' VGU cannot understand such commands. Such commands are interpreted by CPU software

algorithms, as show in Fig. 2 by blocks 4, 5, 6 and 7, which are not relevant points for applicants' claims.

Applicants' display list respects the data order specified in claims 2 - 5, for example, and the VGU hardware circuit process only that specific data sequence and not "high-level graphics commands..." as specified in Koselj patent [20] (inside Summary of the invention) or "vector graphics commands..." [91] (inside Description of the preferred embodiments).

Accordingly, applicants respectfully submit that the claimed VGU is a graphic processor engine not related to interpreter instructions (nor a random sequence of them). The VGU processes only object data. Also the VGU is not a display driver unit as Koselj describes in claim 1, in fact the VGU needs another circuit in order to drive a generic display.

Regarding Koselj Fig. 17 and [11], Koselj does not show an X edge sorting hardware circuit, neither does Koselj explains how an X edge sorting is done. Only in Koselj's Fig. 17 is there a block called Y Sort & Slope Divide, inside an Edge Draw block. Such block apparently combines only to the code line shown in [122], where Koselj mentions only a Y sorting control line:

```
// Always draw from top to bottom (Y Sort)
if (y1<y0)
{
......
}
```

On the contrary applicants' X edge hardware sorting circuit is described inside the specification and it is targeting the X values data (applicants' VGU hardware circuit does not arrange Y data in any means). It is necessary to underline the axes' consistency of both the cited patent and applicants' specification: X axis in Koselj means X axis in applicants' discussion, too. The Y axis processing of Koselj cannot be equated to applicants' X edge hardware sorting circuit.

Moreover, applicants' X edge sorting circuit is made by Dual Port memory and LIFO (Last Input First Output) memory structures which optimize the memory access and the algorithm's execution times (see applicants' hardware sort memories shown in Fig. 5, the associated algorithm flow chart is in Fig. 5 (c) and the full algorithm description is outlined at page 8, for example).

The algorithm has the unique behaving of ordering every new incoming or updated X edge data into the previously ordered X edge memory structure (dual port and LIFO).

Accordingly, it is respectfully submitted that Koselj's Y sorting circuit does not have any relation to nor the capabilities of applicants' X sorting hardware circuit.

The method used to tessellate in Koselj patent is the standard parametric algorithm for drawing Bézier curves. There is no description of the Koselj's hardware architecture implementation. The parametric algorithm software by itself,

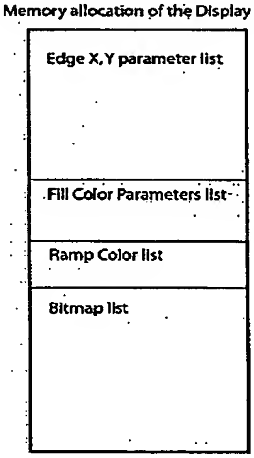
described by Koselj, is the mathematical description of the Bézier curve.

Applicants' hardware Bézier curve subdivision is recited with specific details in applicants' claim 7, 12, 13 and 14.

Koselj's tessellation circuit does not mention any specific hardware circuit or memory, neither the lower time execution that applicants' subdivision hardware is capable of achieving (see this document, regarding claim 7).

Regarding Claims 2-5, applicants' intention was to underline the structure of the display list, therefore the data sequence. The following picture illustrate display list memory allocation:

Memory allocation of the Display List



Regarding claim 6, as stated in this document above, Koselj disclosed a Y (not X) sorting technique which is not related to applicants' X edge sorting hardware circuit. Also the Koselj Y sort hardware circuit is not disclosed, therefore Koselj does not

reveal the applicants' innovative X hardware sorting circuit design.

Regarding claim 7, Fig 4 of the applicants' specification shows the implementation of a unique hardware circuit for the Bézier and De-Casteljau algorithm (displayed in Fig 4b).

Koselj's software algorithm is the Bézier polynomial formula, originally developed by Pierre Bézier in the 1970's. That algorithm as shown by Koselj Fig 8 and [140-149] is not the most efficient way to evaluate the curve subdivision by a hardware circuit and Koselj does not explain how the specific algorithm is done in hardware.

In contrast, applicants' Bézier and De-Casteljau circuit is an innovative hardware implementation of a different algorithm described in page 5 and 6, and in Fig. 4(b) and (c) of the present application.

This new circuit is designed to reduce complexity and optimize especially the execution speed. It is important to note the dual-port memory addressing scheme: this is the most efficient to read and write the control points parameters during the sub-division process.

As applicants state in Page 6, an N=8 segment edge subdivision is executed in only

$$3 + 6 + 12 = 21 \text{ clock cycles, where:}$$

a. 3 clocks for reading and writing the first curve subdivision, obtaining 2 sub curves;

b. 6 clocks for reading and writing the second curve subdivision, obtaining 4 sub curves;

c. 12 clocks for reading and writing the third curve subdivision, reaching the N=8 curve segments edge subdivision.

Or, more generally, an N edge subdivision it is accomplished by applicants' circuit only in the following formula:

$$\#\,\textit{Clock cycle} = K \cdot \sum_{\log_2 N}^{I=1} 2^{(I-1)}$$

where K is the #clock per step and N is the number of edge sub\division.

In our example K is equal to 3 clocks due to a 32-bit access memory bus architecture and a 16 bits X/Y coordinate depth. Maintaining a 16 bits X-Y data depth system, K can be as low as 1 clock when the memory bus is expanded to 96 bits.

Accordingly it is respectfully concluded that applicants' Edge subdivision hardware circuit refers to a different and more innovative algorithm that allows a shorter curve subdivision execution time, compared to the Koselj software polynomial formula.

Regarding claim 8, applicants underscore the unique parallel structure of the anti-aliasing hardware circuit. As applicants

describe at page 11, the AA (anti-aliasing) Buffer circuit is based on a parallel adder group, capable of processing 4 real pixels every execution clock thanks to the integration with a dual port memory, see Fig. 6(a), (b), (c) and (d).

With respect to claim 9, both applicants' and Koselj's circuits process linear and radial gradients. However applicants' circuit use a unique matrix "hardware engine" to transform (resize etc...) the gradients which is not mentioned by Koselj.

Applicants' specification explains how a dedicated hardware circuit optimizes the reading and writing mechanism during the bitmap pixel mapping (see Pg. 9). In particular applicants use a temporary memory that stores the number of replicated source pixel that needs to be drawn in to the destination buffer, Fig 7(a). In contrast, Koselj does not describe the algorithm of reading a bitmap and mapping every original bitmap pixel to the final destination without empty spaces even when the bitmap is scaled or rotated.

Accordingly, it is submitted that applicants use a unique matrix "hardware engine" and temporary memory to transform (resize etc...) the gradients and bitmaps that are not mention or considered by Koselj.

Regarding claim 10, a composing buffer (dump buffer) is used in both Koselj and applicants' specification, and this is another common art technique to reduce accesses to the external memory.

However, Koselj requires to re-send commands for every shape to be drawn bigger than the buffer (see Koselj [168]).

In contrast, applicants' Vector Graphics Circuit uses the same display list containing the same shapes. Therefore by modifying a clipping region parameter in the VGU unit, that can dimensioned to include only the composing buffer, it is possible to draw the same display list and to transfer the buffer over the display memory multiple times without CPU intervention.

Therefore, applicants' dump buffer architecture allows a better speed optimization and a lower use of CPU resource (no re-send object data required).

Referring to claims 11-14, as noted and explained by applicants in this document, applicants' Edge curve subdivision circuit is a different algorithm implementation with respect to Koselj. Therefore we state that claims 12 to 14 are a valid description of how the Applicants' hardware subdivision algorithm can only be achieved in a real hardware circuit.

In view of the above, it is respectfully submitted that applicants' claims 1-14 are neither taught nor suggested by Koselj et al.

In light of the above noted amendments and remarks, this application is believed in condition for allowance and notice thereof is respectfully solicited. The Examiner is asked to contact applicant's attorney at 503-224-0115 if there are any questions.

Respectfully submitted,

James H. Walters, Reg. No. 35,731

Customer number 802
DELLETT AND WALTERS
P.O. Box 2786
Portland, Oregon 97208-2786 US
(503) 224-0115
DOCKET: B-322

Certification of Facsimile Transmission

I hereby certify that this correspondence is being facsimile transmitted to the Patent and Trademark Office on this August 2, 2005.